ИССЛЕДОВАНИЕ ПРОГРАММНОЙ РЕАЛИЗАЦИИ АЛГОРИТМА СЖАТИЯ JPEG С КОНТРОЛЕМ БИТРЕЙТА НА SOC ARM I.MX233

И.О. Осташевский, А.Н. Осокин
Томский политехнический университет
ostashio@gmail.com

Введение

Существует ряд прикладных задач, при решении которых использование современных методов сжатия изображений невозможно или имеет ряд трудностей: патентные ограничения, низкая вычислительная мощность аппаратуры, несовместимость с программным обеспечением существующих систем, небольшая ширина каналов передачи данных и т. д. Решением такой задачи может стать модифицированный и реализованный авторами кодер JPEG с функцией контроля битрейта, обладающий низкими требованиями к ресурсам, для которого декодером является любая стандартная реализация JPEG.

Алгоритм контроля битрейта для кодера JPEG

В работе [1] предложен алгоритм кодера JPEG с возможностью контроля битрейта и на ее основе, в работе [2] предложен алгоритм контроля битрейта для полутоновых изображений, который отличается простотой программной реализации, требует менее двух полных процедур сжатия, обеспечивает приемлемое качество восстановленных изображений и может быть реализован на недорогих миникомпьютерах.

1. Задание битрейта (коэффициент сжатия K) на входе программы;
2. Проведение ДКП согласно стандарту JPEG;
3. Равномерная выборка ДКП-блоков;
4. Квантование выбранных блоков с высоким качеством Q1 ∈[90,95];
5. Сжатие с помощью статических таблиц Хаффмана;
6. Подсчет необходимого количества бит (N) необходимого для представления сжатых блоков и на его основе линейный расчет конечного размера всего изображения (N1);
7. Если N меньше N1, то переход на шаг 9.
8. Вычисление необходимого количества бит для представления сжатых ДКП-блоков;
9. Поиск необходимого коэффициента квантования из интервала Qn∈[5,Q1) методом дихотомии, при условии обеспечения необходимого (близкого к вычисленному на шаге 7) размера ДКП-блоков; При сжатии использовать статические таблицы Хаффмана; Если $|Q_n - Q_{n-1}| \geq 2$, повторить шаг;
10. На основе найденного коэффициента квантования Q_n квантование и сжатие всего изображения с использованием динамических таблиц Хаффмана.

Краткий обзор аппаратного обеспечения, как базы для выполнения алгоритма

Для выполнения алгоритма был проведен анализ рынка одноплатных компьютеров [3] в ценовом диапазоне $30-40. Задача обзора - проанализировать рынок одноплатных компьютеров способных реализовать алгоритм.

Таблица 1. Сравнительные характеристики контроллеров

Модель	iMX233-OLinuXino-NANO	Raspberry Pi, Model B	pcDuino Lite
Производитель	Freescale	Broadcom	Allwinner
Частота ядра (МГц)	454	750	1000
Оперативная память (Mb)	64	512	512
Flash-память (Mb)	0	0	0
Количество USB-Портов	1	2	2
Ethernet (Mbit/s)	-	100	100
Цена $	30	35	39

В результате анализа свойств был выбран миникомпьютер iMX233-OLinuXino-NANO. Миникомпьютер впоследствии был встроен в испытательный стенд.

Испытательный стенд на основе SoC ARM i.MX233 OLinuXino-NANO

Испытательный стенд представляет собой миникомпьютер от фирмы Freescale, а так же набор периферийных устройств, подключенных через USB-Hub 2.0, жестко закрепленных на пласте оргстекла. Рабочий макет обладает определенным набором характеристик:

- Размер 43 × 62 мм;
- Частота ядра ARM9 454 МГц;
- Объем DDR I 64 Мб, частота 200 МГц;
- Диапазон напряжения БП 20 ± 7 В;
- MicroSD для твердотельных накопителей;
- Один USB-хост порт, обеспечивающий питающий ток до 1,5 А.

Набор периферийных устройств, включенных в состав испытательного стенда:

- Последовательный порт RS-232;
- Fast Ethernet адаптер, обеспечивающий подключение к сети интернет до 100 Мб/с;
- Web-камера (Logitech C170 или другая камера поддерживающая разрешение до 1920 на 1080 точек или FullHD).

Испытательный стенд управляется операционной системой Debian 6.0 с ядром 2.6.31. и загружается с MicroSD карты объемом 4 Гб.

Реализация алгоритма на испытательном стенде

Реализация алгоритма была проведена в несколько этапов:

- установка сторонней библиотеки libjpeg[4] на ПК под управлением Ubuntu, а так же на испытательном стенде;
- установка Eclipse CDT для удобной и более привычной разработки;
- создание проекта и написание алгоритма;
- подключение кросс компилятора, и компиляция проекта для выполнения на испытательном стенде;
- запуск и проведение тестов непосредственно на испытательном стенде;
- фиксирование результатов алгоритма на испытательном стенде, и сравнение с результатами подобной работы [2], целью которой являлся расчет эффективности.

Исследование реализации алгоритма по скорости выполнения и эффективности

Для исследования необходима получение числовых данных при работе алгоритма JPEG реализованного на SoC ARM i.MX233. Для исследования необходимы следующие данные:

- Набор полутоновых изображений Calgary Corpus Gray Set 2, включающий в себя 18 полутоновых изображения.
- Коэффициент сжатия K∈[4,30]. Коэффициент ограничен пороговым значением 30, так как изображения с большим коэффициентом получаются в неприемлемом качестве.

Результаты исследования приведены в таблице №2. Алгоритм протестирован на изображении «barb_1.bmp"» из набора «Calgary Corpus Gray Set 2» и результаты сопоставлены с результатами исследования этого же алгоритма на предмет быстродействия, которые в свою очередь сравниваются с результатами работы утилиты cjpeg[5].

Таблица 2. Результаты алгоритма и cjpeg

Коэффициент сжатия (K)	Коэффициент квантования (Q)	Разработанный алгоритм			cjpeg
		Изменение размера, байт/%	Количество сжатий	Время сжатия (мс)	Время сжатия (мс)
4	88	–1590/–2,43	1,52	205	285
5	82	–928/–1,77	1,52	204	284
6	75	–1043/–2,39	1,52	205	286
7	67	–804/–2,15	1,37	202	285
8	58	–557/–1,70	1,44	200	282
9	49	–541/–1,86	1,59	201	283
10	44	399/1,52	1,52	205	284
11	38	382/1,60	1,52	202	281
12	32	–450/–2,06	1,52	200	280
13	29	–234/–1,16	1,52	204	282
14	27	252/1,35	1,52	204	282
15	23	–623/–3,56	1,52	202	283
16	21	–652/–3,98	1,52	202	282
17	21	312/2,02	1,52	201	282
18	18	–633/–4,35	1,52	201	281
19	18	133/0,96	1,52	200	281
20	16	–517/–3,94	1,52	201	281
21	16	107/0,86	1,52	202	281
22	16	675/5,67	1,52	201	282
23	12	–1577/–13,84	1,52	200	281
24	12	–1102/–10,09	1,52	201	280
25	12	–665/–6,34	1,52	202	280
26	12	–262/–2,60	1,52	200	280
27	12	111/1,14	1,52	201	280
28	10	–893/–9,54	1,52	200	280
29	10	–570/–6,31	1,52	200	280
30	10	–269/–3,08	1,52	200	280

Реализован на SoC ARM i.MX233 алгоритм полутоновых изображений для кодера стандарта JPEG с контролем битрейта. Данная реализация способна быстро сжимать изображения, (быстрее стандартной реализации алгоритма JPEG в 1,4 раза), проводя при этом 2 неполных сжатия. В результате, алгоритм способен сжимать изображения в среднем на 10% эффективнее стандартной реализации алгоритма.

Литература

1. R.D. Nguyen Rate control and bit allocation for JPEG transcoding: Master’s thesis. Cambridge: Massachusetts Institute of Technology. 2007. 51с.

2. Д.В. Сидоров, А.Н. Осокин, “Исследование быстродействия модифицированного кодера JPEG на SoC ARM i.MX233с поддержкой контроля битрейта”. Известия ТПУ. 2012 г. 320т. 70-73с.

3. List of single-board computers [Электронный ресурс]. Режим доступа: URL:http://en.wikipedia.org/wiki/List_of_single-board_computers - свободный.

4. Libjpeg [Электронный ресурс]. Режим доступа: URL:http://libjpeg.sourceforge.net/ - свободный.

5. Linux-команды [Электронный ресурс]. Режим доступа: URL:http://linuxcommand.org/man_pages/cjpeg1.html - свободный.